Paper No. 15

UNITED STATES PATENT AND TRADEMARK OFFICE

————————

BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES

————————

Ex parte MARIA C. AZUA and VIKTORS BERSTIS

————————

Appeal No. 2001-1507
Application No. 08/972,129

————————

ON BRIEF

————————

Before HAIRSTON, JERRY SMITH, and DIXON**, Administrative Patent Judges.**

DIXON, **Administrative Patent Judge**.

DECISION ON APPEAL

This is a decision on appeal from the examiner's final rejection of claims 1-28, which are all of the claims pending in this application.

We REVERSE.

BACKGROUND

Appellants' invention relates to a data processing system and method for storing data in a communication network. The system retrieves an application from a network, computes a signature using the application and compares the computed signature to a prior stored signature to determine if the program is to be compiled. An understanding of the invention can be derived from a reading of exemplary claim 1, which is reproduced below.[1]

1. A method for operating a communication network having a user data processor, comprising the steps of:

retrieving an application;

computing a first signature using the application;

comparing the first signature with a second signature previously stored in a first memory of the data processor; and

selectively compiling the selected portion of the application to generate interpreted code when the first signature does not correspond to the second signature.

The prior art references of record relied upon by the examiner in rejecting the appealed claims are:

Zhang, X. Nick, "Secure Code Distribution," COMPUTER, pp. 76-79 (June 1997).

---

[1] We note that independent claim 1 and independent claims 7 and 18 contain the phrase "the selected portion of the application," but we find no express antecedent basis for this limitation. We leave it to the examiner to correct this minor deficiency.

Hsieh, Cheng-Hsueh A., "Optimizing NET Compilers for Improved Java Performance," <u>COMPUTER</u>, pp. 67-75 (June 1997).

Claims 1-28 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Hsieh in view of Zhang.

Rather than reiterate the conflicting viewpoints advanced by the examiner and appellants regarding the above-noted rejections, we make reference to the examiner's answer (Paper No. 12, mailed Sep. 11, 2000) for the examiner's reasoning in support of the rejections, and to appellants' brief (Paper No. 9, filed Jan. 27, 2000), appellants' supplemental brief (Paper No. 11, filed Jun. 26, 2000) and reply brief (Paper No. 13, filed Nov. 17, 2000) for appellants' arguments thereagainst.

OPINION

In reaching our decision in this appeal, we have given careful consideration to appellants' specification and claims, to the applied prior art references, and to the respective positions articulated by appellants and the examiner. As a consequence of our review, we make the determinations which follow.

Appellants argue at great length and in great depth throughout the briefs that appellants find no support for the examiner's position that Zhang teaches or suggests computing a signed object for security purposes and shows updating of the signed object by using the SignatureBlock and the HashBlock. (For example, see

supplemental brief at page 5-9.) We agree with appellants. From our review of the

teachings of Zhang, we find that Zhang teaches the authentication of the downloaded

Applets which have an embedded signature, but we do not find that the embedded

signature is compared to a stored signature. Zhang discloses the comparison of a

decrypted data value with a locally generated hash result. Appellants argue that the

examiner's rejection is based upon conclusions by the examiner and not upon evidence

and therefore, based upon hindsight.   (See brief at pages 9-10.) We do not find that

the portions of Zhang relied upon by the examiner at pages 4-5 of the answer support

the examiner's position, and we agree with appellants.

Appellants argue that the examiner's reliance on the teachings of Hsieh are

misplaced. (See brief at page 10 et seq.) We agree with appellants and find that Hsieh

teaches merely a comparison between an actual compiling or interpreter-based JAVA

VM and a translation using Native Language Translation (NET) compiler. Hsieh also

discloses that the verifier determines whether or not the native code copies of the byte

code files exist in the disk cache and if found are used for execution and if not found

then the code is generated and a copy stored for future use. (See Hsieh at page 69,

col. 2.) While Hsieh discloses the translation or compiling of the codes, we do not find

that this is determined as a result of a signature comparison as recited in the language

of independent claim 1.

The examiner maintains that Zhang teaches retrieving an application and computing a signature using the application. (See answer at page 14.) The examiner maintains that Zhang also teaches the limitation of comparing with respect to the "verification" using the hash of Zhang. The examiner then maintains that appellants' disclosure at pages 21-22 with respect to the disclosed comparing clearly corresponds to the hash of Zhang. We disagree with the examiner and do not find that the hash of the signature of Zhang is compared to prior stored signatures. Rather the hash result which is generated is verified and the signature is compared to the prior signatures, but the first signature would not then be generated by the application. Therefore, the examiner's interpretation of Zhang is strained to try to manipulate the processes of Zhang to meet the claim limitations. We find the examiner's interpretation flawed. Alternatively, the examiner maintains that "all Java byte code, Java Applets are down loaded for execution." (See answer at page 14.) While the compilation may be intended for all code, Zhang implies that if the code does not verify, then there may be a security problem and may not compile the code for safety reasons. Therefore, this finding does not appear to be true for all code and the examiner's position is not persuasive.

Alternatively, the examiner maintains at page 15 of the answer that Hsieh

teaches the same method of selective compilation except for the use of a signature.

We agree with the examiner that Hsieh teaches compilation and is silent as to the use

of a signature. While Hsieh discloses the use of a verification process at page 72 and

73, we do not agree with the examiner that Hsieh teaches a comparing step

corresponding to the appellants' description. (See answer at page 15.) Hsieh discloses

that "classes are validated during the translation, leaving the native code on the local

machine for future invocations of the program. This verification is performed again only

when the updater spots a change in a class." (Hsieh at pages 72-73.) From our review

of the teachings of Hsieh, we find no teaching or suggestion of how the updater

determines or spots a change in a class. The examiner relies on page 69 and Fig. 2 of

Hsieh to teach that a cache copy is used if available and otherwise the code is

translated. Here again, the examiner strains to stretch the teachings of Hsieh in an

attempt to meet the language of independent claim 1.   Since there is no signature

generated by the application, there can be no suggestion to compare this signature to a

stored signature. Therefore, we find no support for the examiner's conclusion at page

15 of the answer that Hsieh's verification is the same as the disclosed verification. The

examiner then concludes that "Zhang inherently teaches the claimed invention despite

compilation, and Hsieh et al., inherently teach the claimed invention despite signature."

(See answer at page 16.) We disagree with the examiner and find that the examiner is

attempting to reconstruct appellants' claimed invention using impermissible hindsight to

pick and choose various discrete components of appellants' claimed invention and

generalize therefrom to meet the language of independent claim 1. The examiner

addresses the appellants' hindsight argument at pages 16-18 and relies on many items

being well known and other items being taught by each of the references and

corresponding to the level of description in appellants' specification. The examiner

maintains that since both references disclose Java byte code which is disclosed in

appellants' specification and since both references both suggest missing elements

which are missing in the other reference that it would have been obvious to one of

ordinary skill in the art at the time of the invention to combine the teaching for

enhancing execution time of the Java byte code and also conforming to security policy

enforcement.   We disagree with the examiner's conclusion and rationale and find that it

is not supported by the teachings in the references nor by a convincing line of

reasoning. Therefore, we find that the examiner has not established a ***prima facie***

case of obviousness of the claimed invention, and will not sustain the rejection of

independent claim 1 and its dependent claims 2-6.

We find similar limitations in independent claims 7, 13, 18, and 24, and we will

not sustain the rejection of these claims and their dependent claims.

CONCLUSION

To summarize, the decision of the examiner to reject claims 1-28 under 35

U.S.C. § 103(a) is reversed.

REVERSED

KENNETH W. HAIRSTON Administrative Patent Judge	)
	)
	)
	)
	) BOARD OF PATENT
JERRY SMITH Administrative Patent Judge	) APPEALS
	) AND
	) INTERFERENCES
	)
	)
	)
JOSEPH L. DIXON Administrative Patent Judge	)

JLD/vsh

JAMES J. MURPHY
5400 RENAISSANCE TOWER
1201 ELM STREET
DALLAS, TX 75270-2199